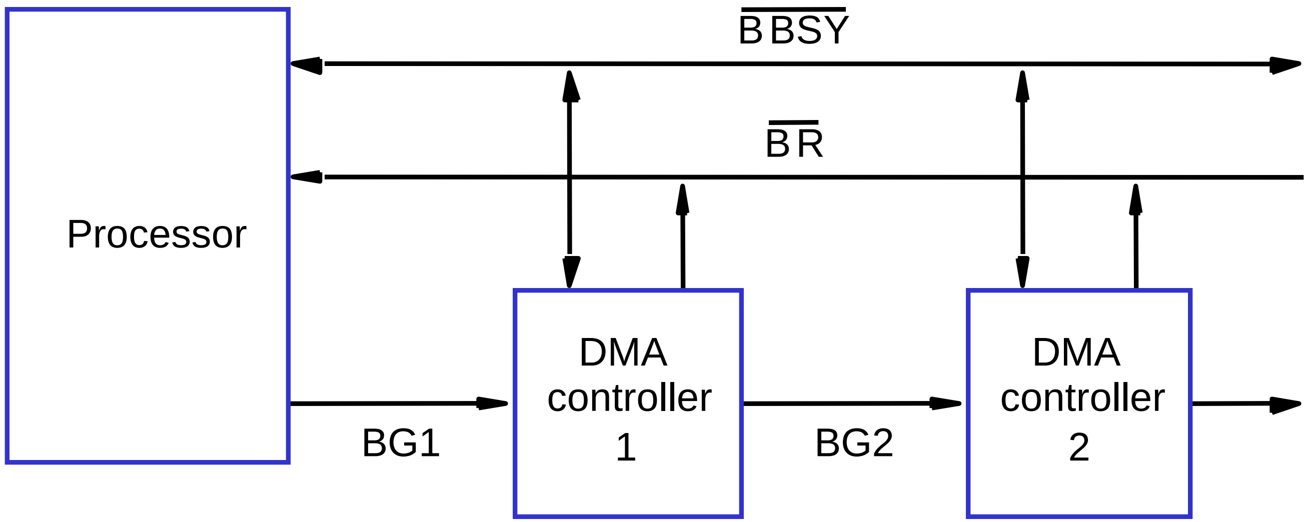
Lecture 5 (part 2)

Topics covered:
Input/Output Organization

Bus arbitration

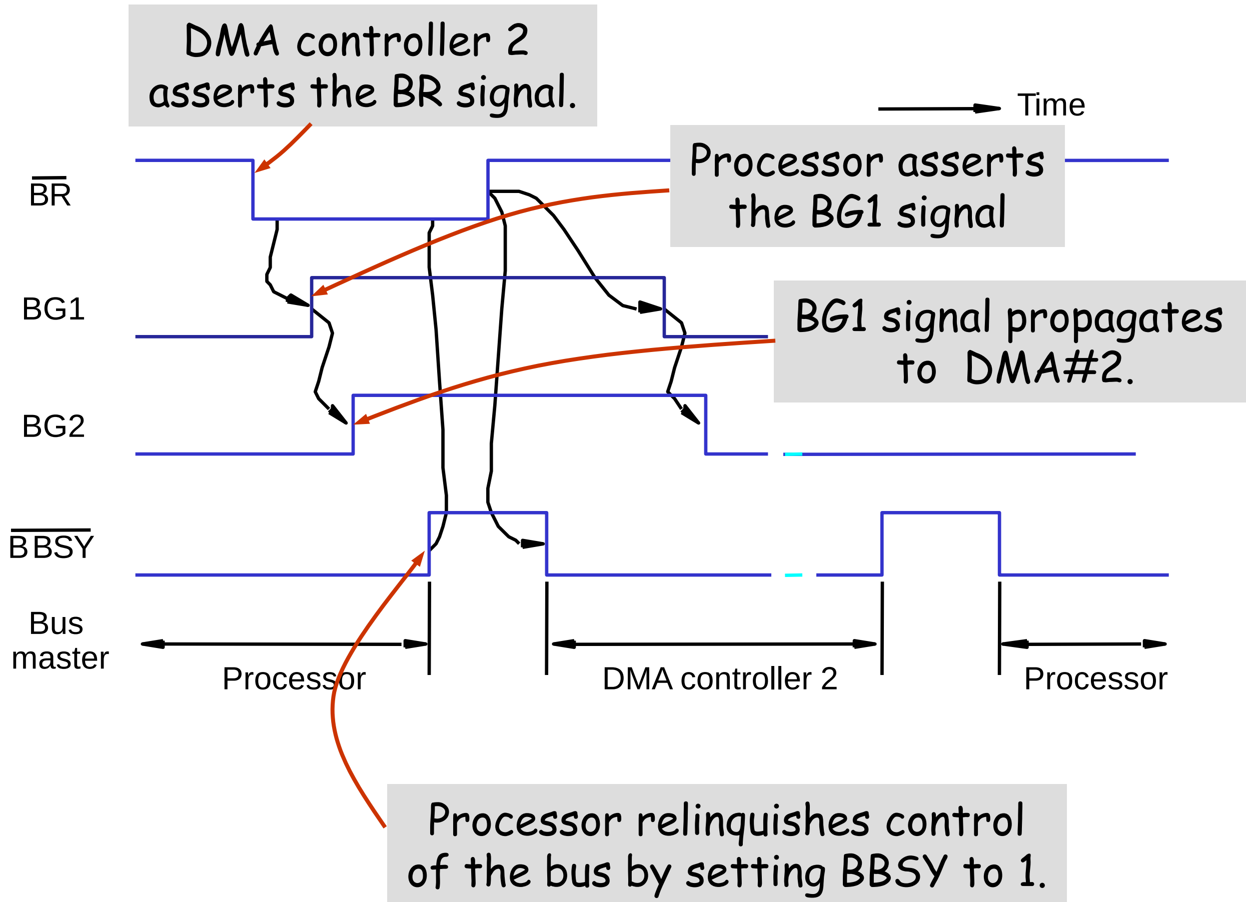
- Processor and DMA controllers both need to initiate data transfers on the bus and access main memory.
- The device that is allowed to initiate transfers on the bus at any given time is called the bus master.
- When the current bus master releases control of the bus, another device can acquire the status of the bus master..
  - The process by which the next device to become the bus master is selected and bus mastership is transferred to it is called <u>bus arbitration</u>.
- Centralized arbitration:
  - A single bus arbiter performs the arbitration.
- Distributed arbitration:
  - All devices participate in the selection of the next bus master.

Centralized arbitration



- *Bus arbiter may be the processor or a separate unit connected to the bus.*
- *Normally, the processor is the bus master, unless it grants bus membership to one of the DMA controllers.*
- *DMA controller requests the control of the bus by asserting the Bus Request (BR) line.*
- *In response, the processor activates the Bus-Grant1 (BG1) line, indicating that the controller may use the bus when it is free.*
- *BG1 signal is connected to all DMA controllers in a daisy chain fashion.*
- *BBSY signal is 0, it indicates that the bus is busy. When BBSY becomes 1, the DMA controller which asserted BR can acquire control of the bus.*

Centralized arbitration (contd..)



Centralized arbitration (contd..)

- Centralized arbitration scheme with one Bus-Request (BR) line and one Bus-Grant (BG) line forming a daisy chain.
- Several pairs of BR and BG lines are possible, perhaps one per device as in the case of interrupts.
- Bus arbiter has to ensure that only one request is granted at any given time.
- It may do so according to a fixed priority scheme, or a rotating priority scheme.
- Rotating priority scheme:
  - There are four devices, and initial priority is 1,2,3,4.
  - After the request from device 1 is granted, the priority changes to 2,3,4,1.

Distributed arbitration

- All devices waiting to use the bus share the responsibility of carrying out the arbitration process.
  - Arbitration process does not depend on a central arbiter and hence distributed arbitration has higher reliability.
- Each device is assigned a 4-bit ID number.
- All the devices are connected using 5 lines, 4 arbitration lines to transmit the ID, and one line for the Start-Arbitration signal.
- To request the bus a device:
  - Asserts the Start-Arbitration signal.
  - Places its 4-bit ID number on the arbitration lines.
- The pattern that appears on the arbitration lines is the logical-OR of all the 4-bit device IDs placed on the arbitration lines.

Distributed arbitration (contd..)

- *Device A has the ID 5 and wants to request the bus:*
  - *Transmits the pattern 0101 on the arbitration lines.*
- *Device B has the ID 6 and wants to request the bus:*
  - *Transmits the pattern 0110 on the arbitration lines.*
- *Pattern that appears on the arbitration lines is the logical OR of the patterns:*
  - *Pattern 0111 appears on the arbitration lines.*

Arbitration process:

- *Each device compares the pattern that appears on the arbitration lines to its own ID, starting with MSB.*
- *If it detects a difference, it transmits 0s on the arbitration lines for that and all lower bit positions.*
- *Device A compares its ID 5 with a pattern 0101 to pattern 0111.*
- *It detects a difference at bit position 0, as a result, it transmits a pattern 0100 on the arbitration lines.*
- *The pattern that appears on the arbitration lines is the logical-OR of 0100 and 0110, which is 0110.*
- *This pattern is the same as the device ID of B, and hence B has won the arbitration.*

Buses

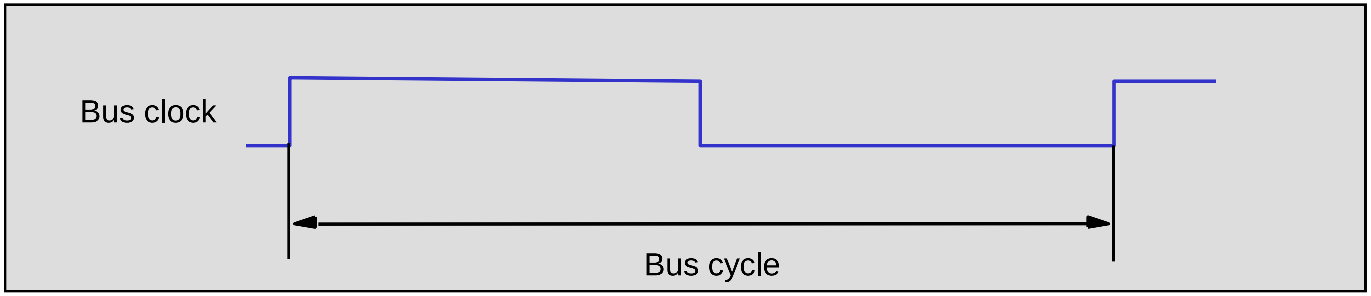
- Processor, main memory, and I/O devices are interconnected by means of a bus.
- Bus provides a communication path for the transfer of data.
  - Bus also includes lines to support interrupts and arbitration.
- A bus protocol is the set of rules that govern the behavior of various devices connected to the bus, as to when to place information on the bus, when to assert control signals, etc.

Buses (contd..)

- Bus lines may be grouped into three types:
  - Data
  - Address
  - Control
- Control signals specify:
  - Whether it is a read or a write operation.
  - Required size of the data, when several operand sizes (byte, word, long word) are possible.
  - Timing information to indicate when the processor and I/O devices may place data or receive data from the bus.
- Schemes for timing of data transfers over a bus can be classified into:
  - Synchronous,
  - Asynchronous.

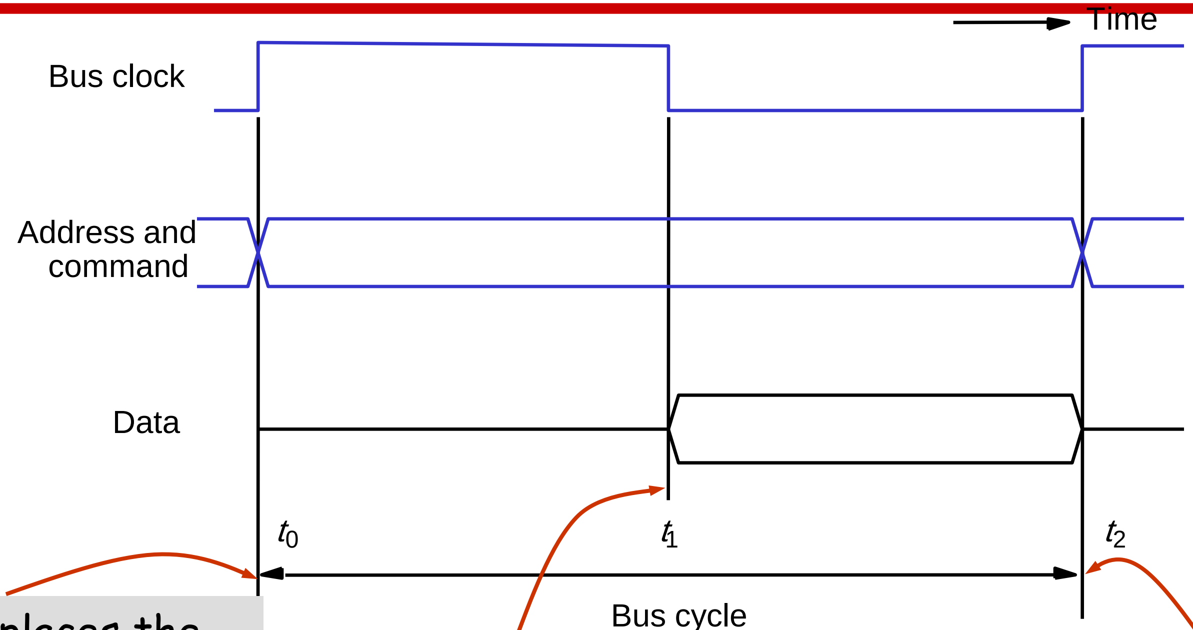
Synchronous bus

- All devices derive timing information from a common clock line.
- The clock line has equally spaced pulses which define equal time intervals.
  - In a simple synchronous bus, each of these pulses constitutes a bus cycle.
- One data transfer can take place during one bus cycle.



Synchronous bus (contd..)

Read operation ideal case



Master places the device address and command on the bus, and indicates that it is a Read operation.

Addressed slave places data on the data lines

Master "strobes" the data on the data lines into its input buffer, for a Read operation.

- *In case of a Write operation, the master places the data on the bus along with the address and commands at time t_0.*
- *The slave strobes the data into its input buffer at time t_2.*

Synchronous bus (contd..)

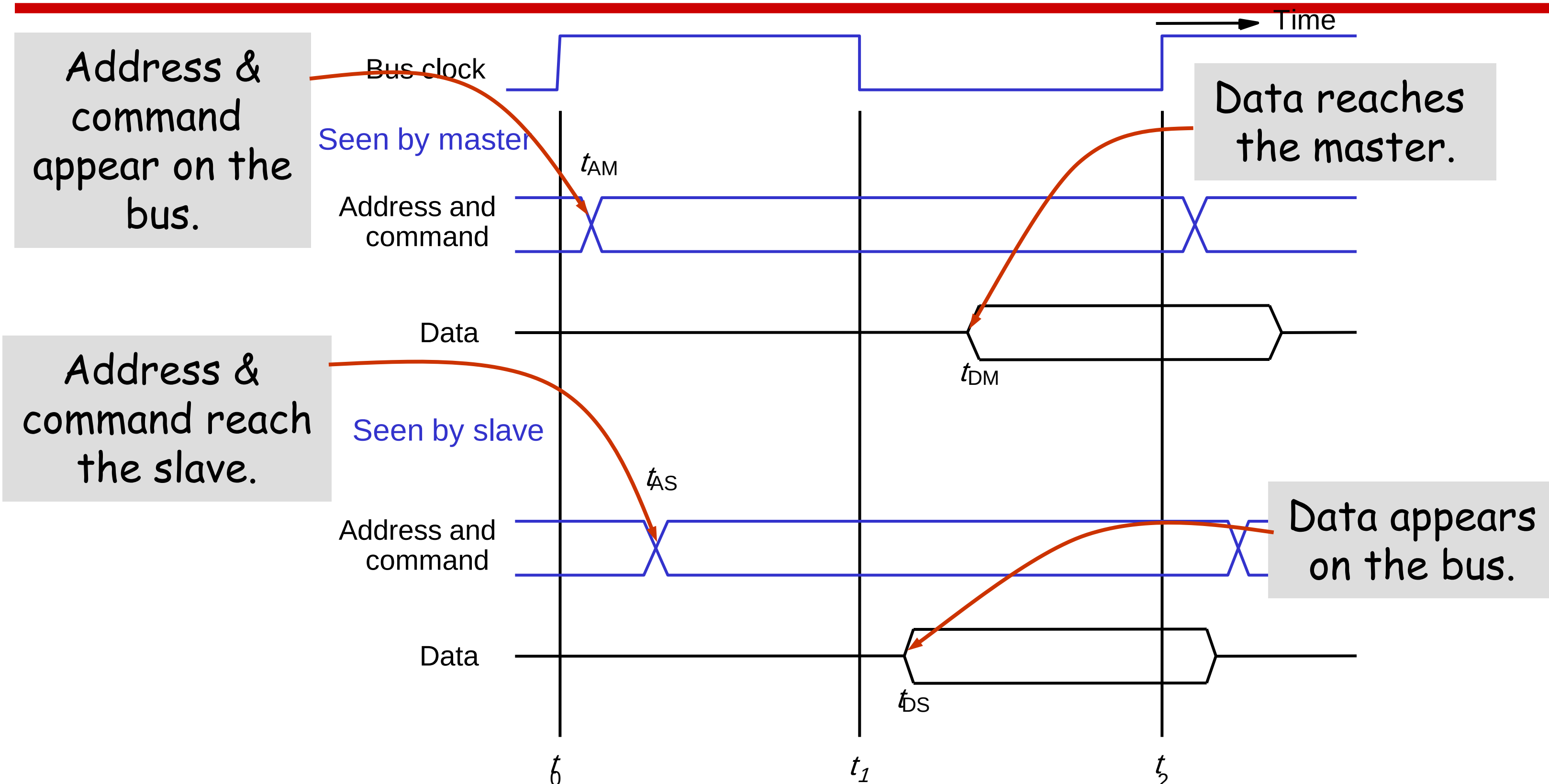
- Once the master places the device address and command on the bus, it takes time for this information to propagate to the devices:
  - This time depends on the physical and electrical characteristics of the bus.
- Also, all the devices have to be given enough time to decode the address and control signals, so that the addressed slave can place data on the bus.
- Width of the pulse $t_1 - t_0$ depends on:
  - Maximum propagation delay between two devices connected to the bus.
  - Time taken by all the devices to decode the address and control signals, so that the addressed slave can respond at time t_1.

Synchronous bus (contd..)

- At the end of the clock cycle, at time t_2, the master strobes the data on the data lines into its input buffer if it's a Read operation.
  - "Strobe" means to capture the values of the data and store them into a buffer.
- When data are to be loaded into a storage buffer register, the data should be available for a period longer than the setup time of the device.
- Width of the pulse $t_2 - t_1$ should be longer than:
  - Maximum propagation time of the bus plus
  - Set up time of the input buffer register of the master.

Synchronous bus (contd..)

Read operation practical case



- *Signals do not appear on the bus as soon as they are placed on the bus, due to the propagation delay in the interface circuits.*
- *Signals reach the devices after a propagation delay which depends on the characteristics of the bus.*
- *Data must remain on the bus for some time after t_2 equal to the hold time of the buffer.*

the period $t_2 - t_{DM}$ is the setup time for the master's input buffer.

Synchronous bus (contd..)

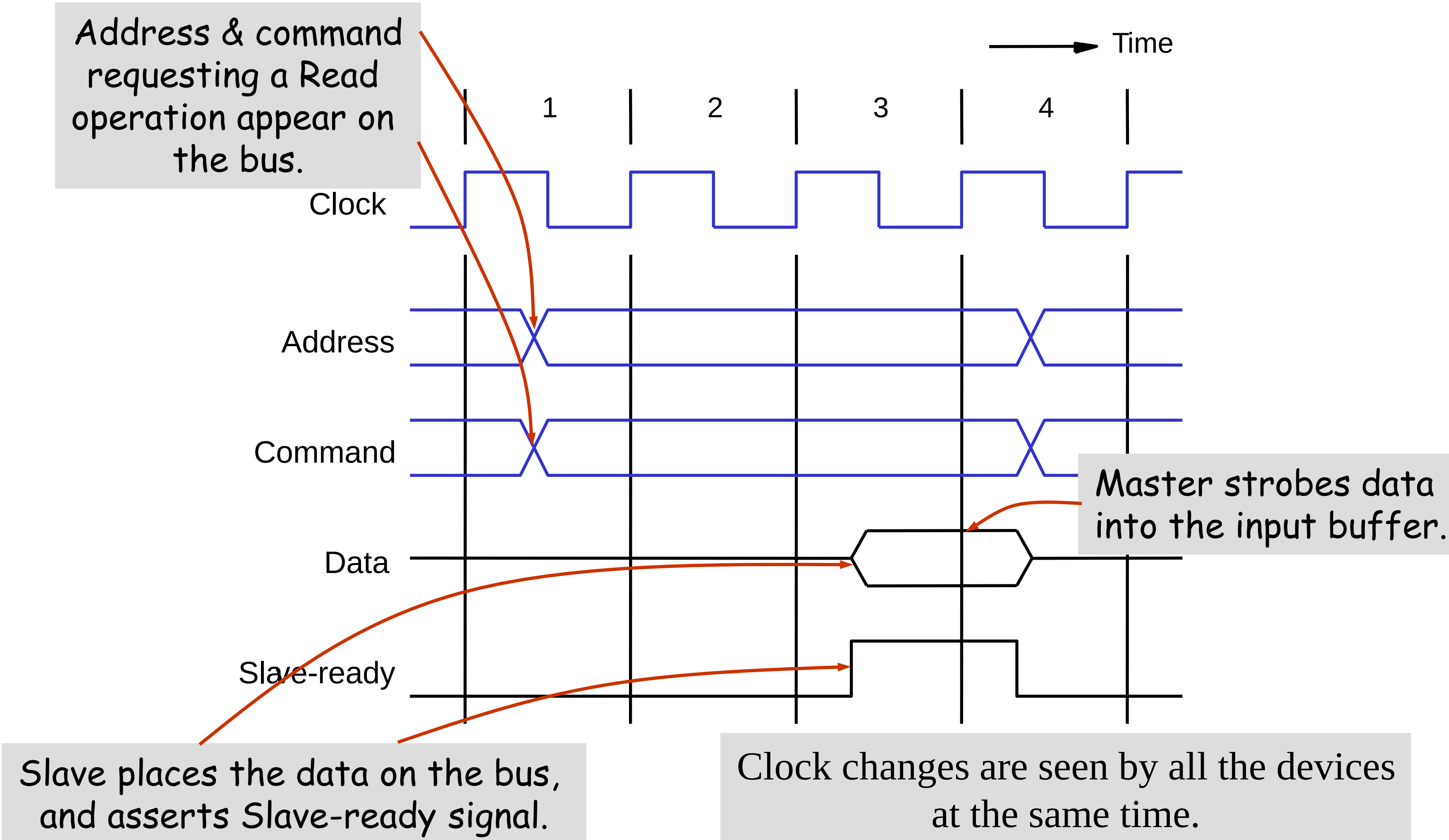
- Data transfer has to be completed within one clock cycle.
  - Clock period t_2 - t_0 must be such that the longest propagation delay on the bus and the slowest device interface must be accommodated.
  - Forces all the devices to operate at the speed of the slowest device.
- Processor just assumes that the data are available at t_2 in case of a Read operation, or are read by the device in case of a Write operation.
  - What if the device is actually failed, and never really responded?

Synchronous bus (contd..)

- Most buses have control signals to represent a response from the slave.
- Control signals serve two purposes:
  - Inform the master that the slave has recognized the address, and is ready to participate in a data transfer operation.
  - Enable to adjust the duration of the data transfer operation based on the speed of the participating slaves.
- High-frequency bus clock is used:
  - Data transfer spans several clock cycles instead of just one clock cycle as in the earlier case.

Synchronous bus (contd..)

An input transfer using multiple clock cycles.



Clock changes are seen by all the devices at the same time.

Synchronous bus (contd..)

- Clock signal used on the bus is not necessarily the same as the processor clock.
  - Processor clock is much faster than the bus clock.
- Modern processor clocks are typically above 500 MHz.
- Clock frequencies used on the memory and I/O buses may be in the range of 50 to 150 MHz.

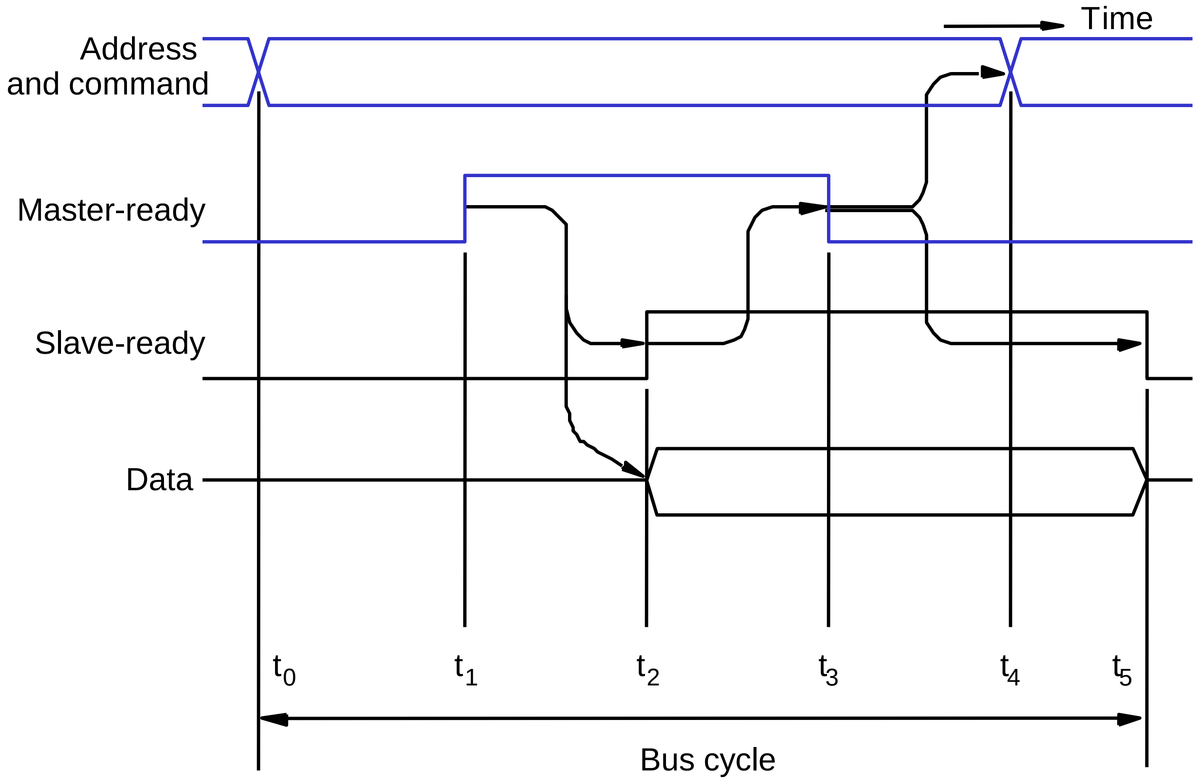
Asynchronous bus

- Data transfers on the bus is controlled by a handshake between the master and the slave.
- Common clock in the synchronous bus case is replaced by two timing control lines:
  - Master-ready,
  - Slave-ready.
- Master-ready signal is asserted by the master to indicate to the master that it is ready to participate in a data transfer.
- Slave-ready signal is asserted by the slave in response to the master-ready from the master, and it indicates to the master that the slave is ready to participate in a data transfer.

Asynchronous bus (contd..)

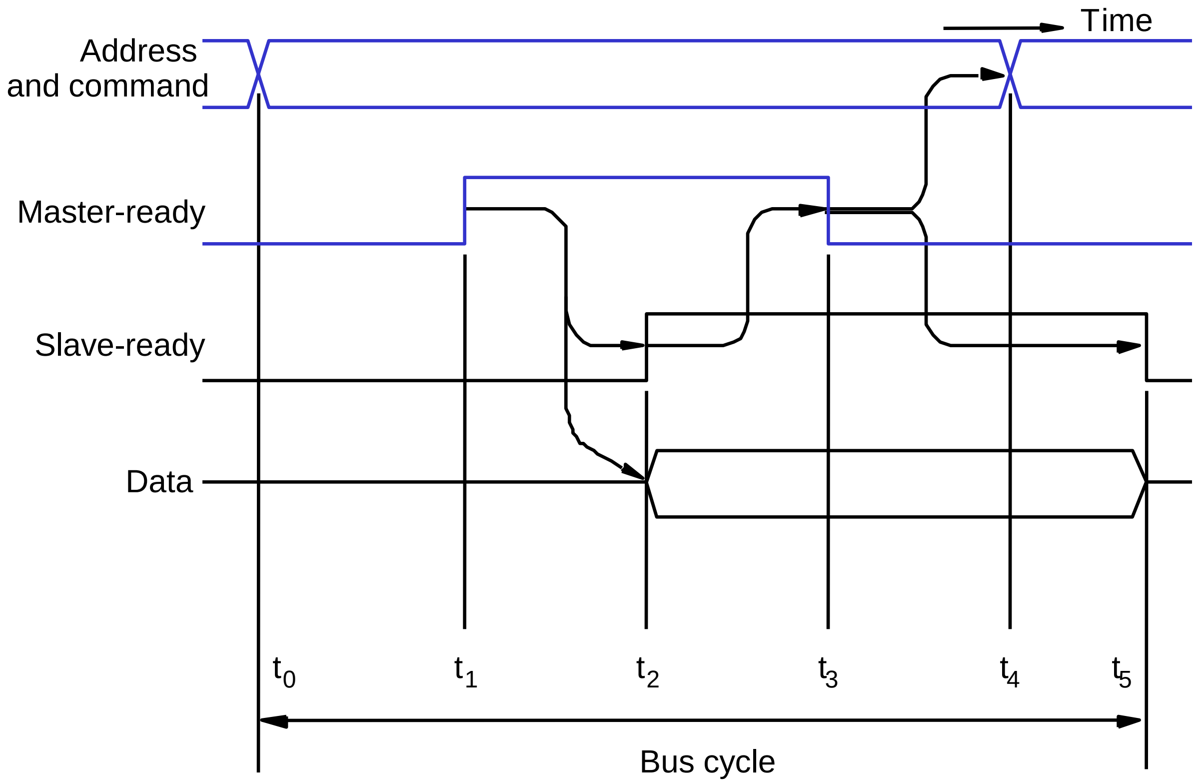
- Data transfer using the handshake protocol:
  - Master places the address and command information on the bus.
  - Asserts the Master-ready signal to indicate to the slaves that the address and command information has been placed on the bus.
  - All devices on the bus decode the address.
  - Address slave performs the required operation, and informs the processor it has done so by asserting the Slave-ready signal.
  - Master removes all the signals from the bus, once Slave-ready is asserted.
  - If the operation is a Read operation, Master also strobes the data into its input buffer.

Asynchronous bus (contd..)



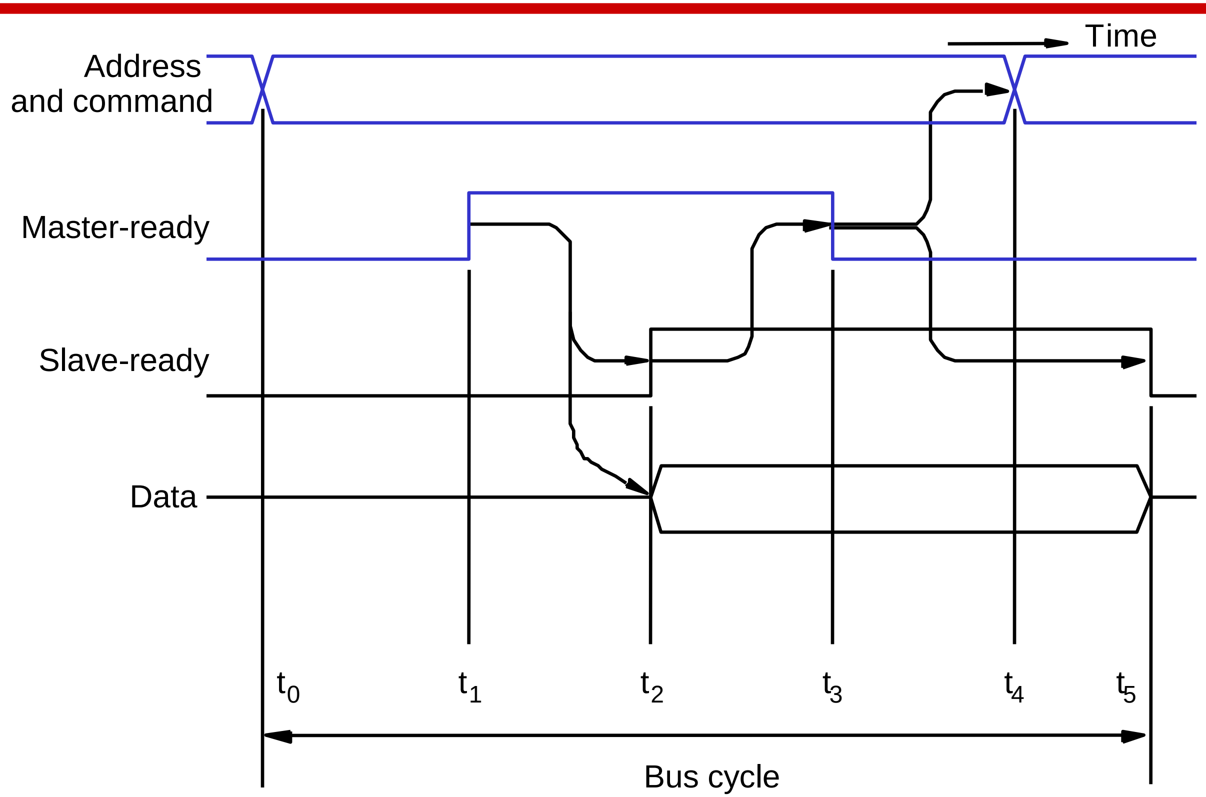
t_0 - Master places the address and command information on the bus. Command indicates that it is a Read operation, that is the data are transferred from the device to the memory.

Asynchronous bus (contd..)



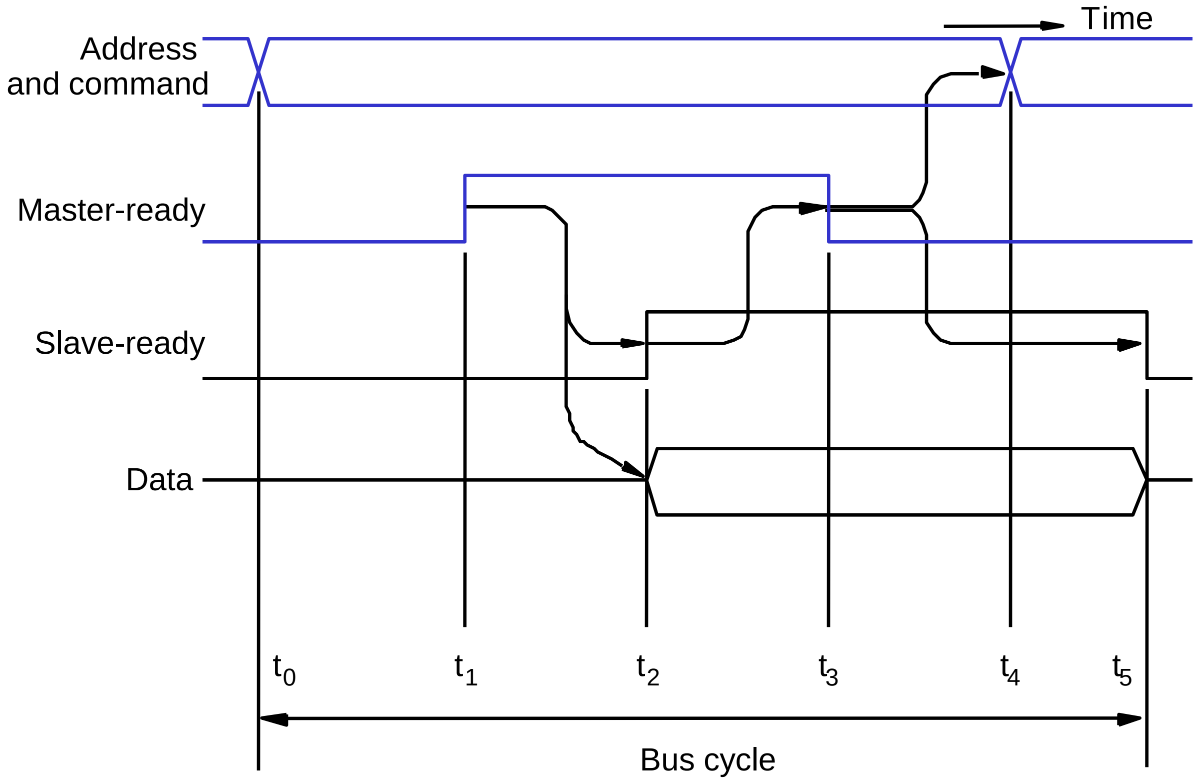
t_1 - Master asserts the Master-ready signal. Master-ready signal is asserted at t_1 instead of t_0 to allow for bus skew. Bus skew occurs when two signals transmitted simultaneously reach the destination at different times. This may occur because different bus lines may have different speeds. t_1-t_0 should be greater than maximum skew. t_1-t_0 should also include the time taken to decode the address information.

Asynchronous bus (contd..)



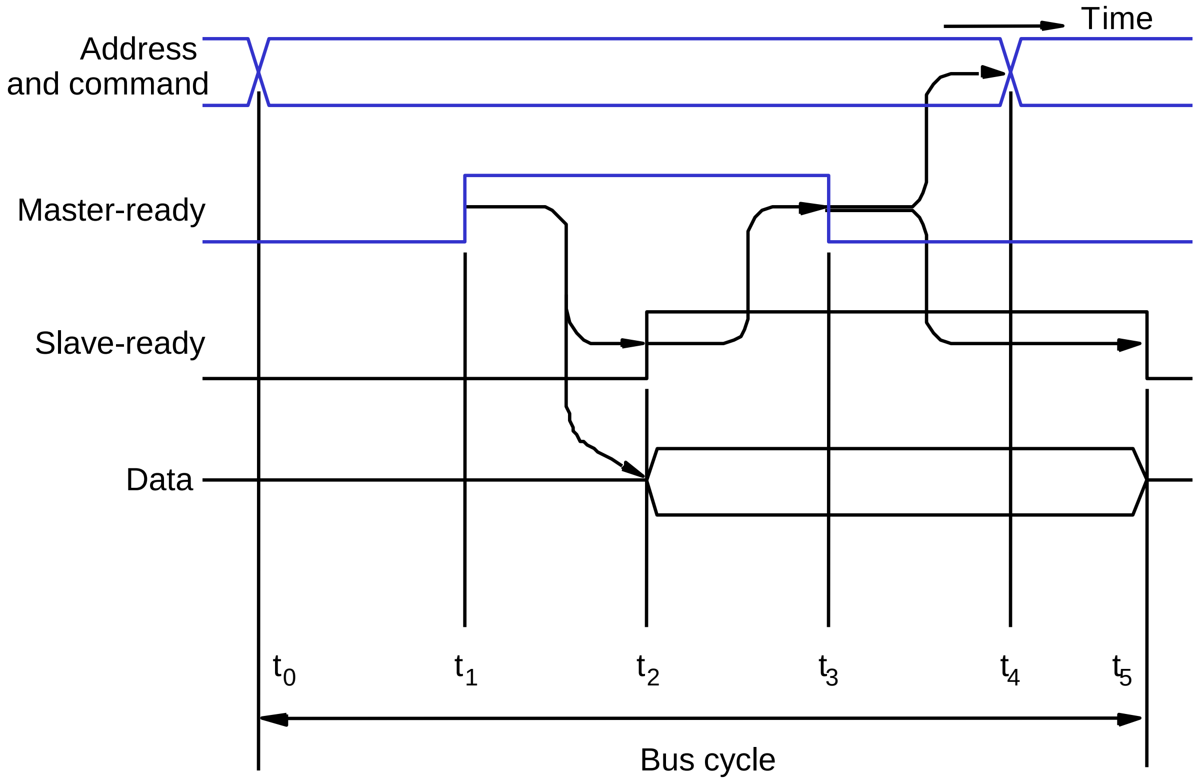
t_2 - Addressed slave places the data on the bus and asserts the Slave-ready signal. The period t_2-t_1, depends on the propagation delay between the master and the slave, and the delay in the slave's interface circuit.

Asynchronous bus (contd..)



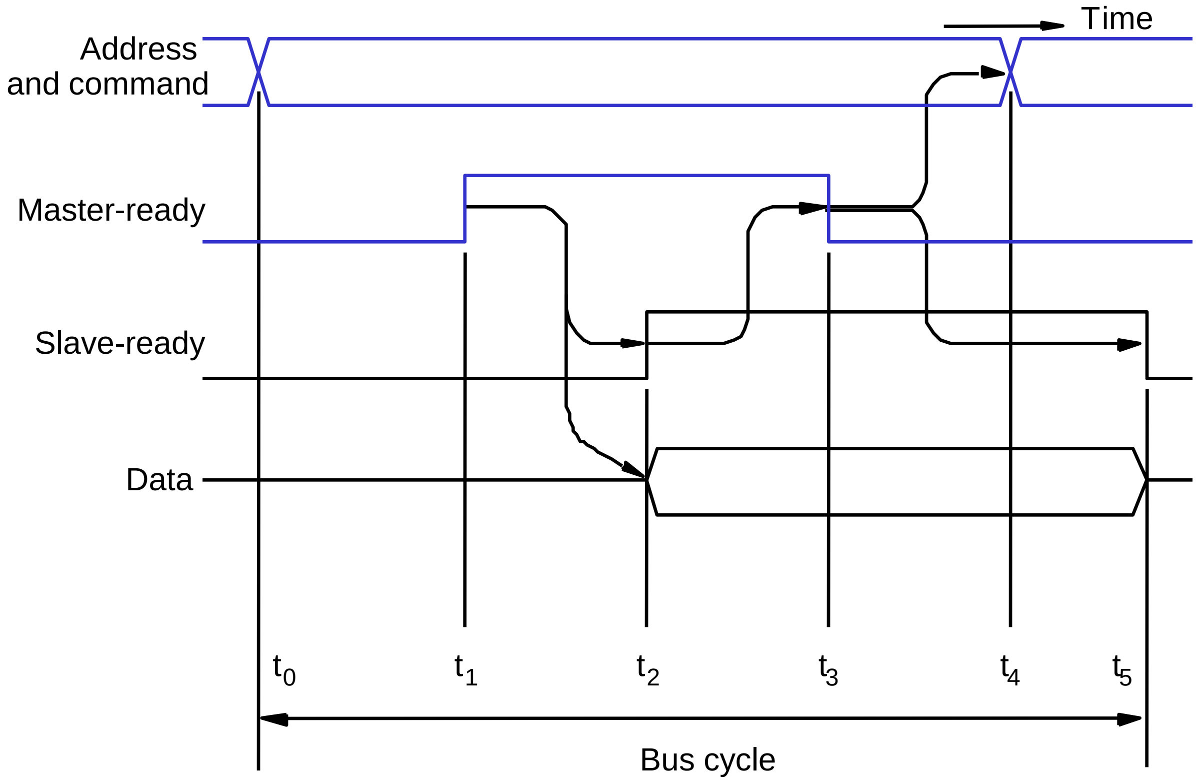
t_3 - Slave-ready signal arrives at the master. Slave-ready signal was placed on the bus at the same time that data were placed on the bus. As a result, the data may also experience bus skew. The master should wait for maximum bus skew plus the setup time of its input buffer and then strobes the data. It also deactivates the Master-ready signal to indicate that it has received the data.

Asynchronous bus (contd..)



t_4 - Master removes the address and command information. t_4-t_3 allows for bus skew. Once Master-ready signal is set to 0, it should reach all the devices before the address and command information is removed from the bus.

Asynchronous bus (contd..)



t_5 - Slave receives the transition of the Master-ready signal from 1 to 0. It removes the data and the Slave-ready signal from the bus.

Asynchronous vs. Synchronous bus

- Advantages of asynchronous bus:
  - Eliminates the need for synchronization between the sender and the receiver.
  - Can accommodate varying delays automatically, using the Slave-ready signal.
- Disadvantages of asynchronous bus:
  - Data transfer rate with full handshake is limited by two-round trip delays.
  - Data transfers using a synchronous bus involves only one round trip delay, and hence a synchronous bus can achieve faster rates.

Assignment

- Parallel ports
- Serial ports

Standard Input /output interfaces

- PCI
- USB
- SCSI

Organize yourself in groups (5 per group) to prepare a presentation and a report about the above topic.